CUNY SPS DATA 621 - CTG5 - HW1

Betsy Rosalen, Gabrielle Bartomeo, Jeremy O'Brien, Lidiia Tronina, Rose Koh

February 27, 2019

Contents

1 DATA EXPLORATION

Professionals and gamblers alike are always seeking to optimize their chances of winning, whether it be sports, games, or their bets on them. Major League Baseball is a multibillion dollar industry where individual teams, players, and those who profit off of their success stand to benefit most from such optimization.

Data from 1871 to 2006 was collected in order to infer how many wins could be expected from the 162 games in a baseball team's season. Each observation represents a season for an unnamed team, and we have a total of 2,276 observations. For each team the target variable, `TARGET_WINS`, represents the number of wins in a given year and has a maximum value of 162 possible wins. In addition to that 15 continuous integer predictor variables were collected (not including the index) representing each team's: base hits, doubles, triples, homeruns, walks, and strikeouts by batters, batters hit by pitches, bases stolen by batters and the number of times they were caught stealing, the number of errors, double plays, walks, hits, and homeruns allowed, and strikeouts by pitchers. The testing data contains the same 15 predictor variables and no target variable so it will be impossible to check the accuracy of our predictions from the testing data.

VARIABLE NAME	DEFINITION	THEORETICAL EFFECT ON WINS
TARGET_WINS	Number of wins	outcome variable
BATTING_H	Base Hits by batters (1B,2B,3B,HR)	Positive Impact
BATTING_2B	Doubles by batters (2B)	Positive Impact
BATTING_3B	Triples by batters (3B)	Positive Impact
BATTING_HR	Homeruns by batters (4B)	Positive Impact
BATTING_BB	Walks by batters	Positive Impact
BATTING_HBP	Batters hit by pitch (get a free base)	Positive Impact
BATTING_SO	Strikeouts by batters	Negative Impact
BASERUN_SB	Stolen bases	Positive Impact
BASERUN_CS	Caught stealing	Negative Impact
FIELDING_E	Errors	Negative Impact
FIELDING_DP	Double Plays	Positive Impact
PITCHING_BB	Walks allowed	Negative Impact
PITCHING_H	Hits allowed	Negative Impact
PITCHING_HR	Homeruns allowed	Negative Impact
PITCHING_SO	Strikeouts by pitchers	Positive Impact

1.1 Summary Statistics

Looking at the Table 2, it can be easily noted that there are outliers present in more than one variable, with `PITCHING_H` being the worst offender. Even at three times the standard deviation, its maximum value lays far outiside of the 68-95-99.7 rule. `FIELDING_E`, on the other hand, has the curious case of having a large difference between its mean and median, indicating there is skew present in this variable as well before any charts are actively looked at. Skewed variables cause bias in linear models and need treatment before being used.

1.2 Shape of Predictor Distributions

Figure. 1 shows the distribution of most of the variables seems normal although `BASERUN_SB`, `BASERUN_CS`, and `BATTING_3B` have a slight to moderate right skew, `FIELDING_E`, `PITCHING_BB`, `PITCHING_H`, and `PITCHING_SO` have an extreme right skew, and `BATTING_HR`, `BATTING_SO`, and `PITCHING_HR` are bimodal.

Table 2: Summary statistics

	n	min	mean	median	max	sd
TARGET_WINS	2276	0	80.79086	82.0	146	15.75215
BATTING_H	2276	891	1469.26977	1454.0	2554	144.59120
BATTING_2B	2276	69	241.24692	238.0	458	46.80141
BATTING_3B	2276	0	55.25000	47.0	223	27.93856
BATTING_HR	2276	0	99.61204	102.0	264	60.54687
BATTING_BB	2276	0	501.55888	512.0	878	122.67086
BATTING_SO	2174	0	735.60534	750.0	1399	248.52642
BASERUN_SB	2145	0	124.76177	101.0	697	87.79117
BASERUN_CS	1504	0	52.80386	49.0	201	22.95634
BATTING_HBP	191	29	59.35602	58.0	95	12.96712
PITCHING_H	2276	1137	1779.21046	1518.0	30132	1406.84293
PITCHING_HR	2276	0	105.69859	107.0	343	61.29875
PITCHING_BB	2276	0	553.00791	536.5	3645	166.35736
PITCHING_SO	2174	0	817.73045	813.5	19278	553.08503
FIELDING_E	2276	65	246.48067	159.0	1898	227.77097
FIELDING_DP	1990	52	146.38794	149.0	228	26.22639

As a result some data transformation will most likely be necessary to improve the accuracy of our model. The standard deviation of the various variables also hints at the intense skewing of some of the variables.

1.3 Outliers

The Figure. 2 shows that there are also a large number of outliers that need to be accounted for, most prevalently in `FIELDING_E` and `BATTING_H` based off of the boxplots below. One such extreme outlier removed implied that there were, on average per game in a single season, 186 hits allowed by pitchers. This is an unrealistic figure, even for those for whom baseball is outside of their realm of understanding.

1.4 Missing Values

The Figure. 3 displays of all the observations gathered across these fifteen variables, there are 3,478 missing values out of 36,416 total data points, which represents 10.187% of the data. Batters hit by pitches was missing the most, with 2,085 instances of missing information, which represents 91.61% of that variable missing. Additionally `Pitching_SO` and `Batting_SO` are missing exact same proportion 4.48% and are missing in the same observations. This data may not be missing at random and so there may be cause for removing it.

1.5 Linearity

Each variable was plotted against the target variable in order to determine at a glance which had the most potential linearity before the dataset was modified.

As can be observed at Figure. 4, the most influential variables are the ones previously discussed to have severe outliers and skew, and their linear relationship is negative - the higher the variable, the lower the target wins. On the other hand, `BATTING_H`, `BATTING_BB` and `BATTING_2B` showed the most promise.

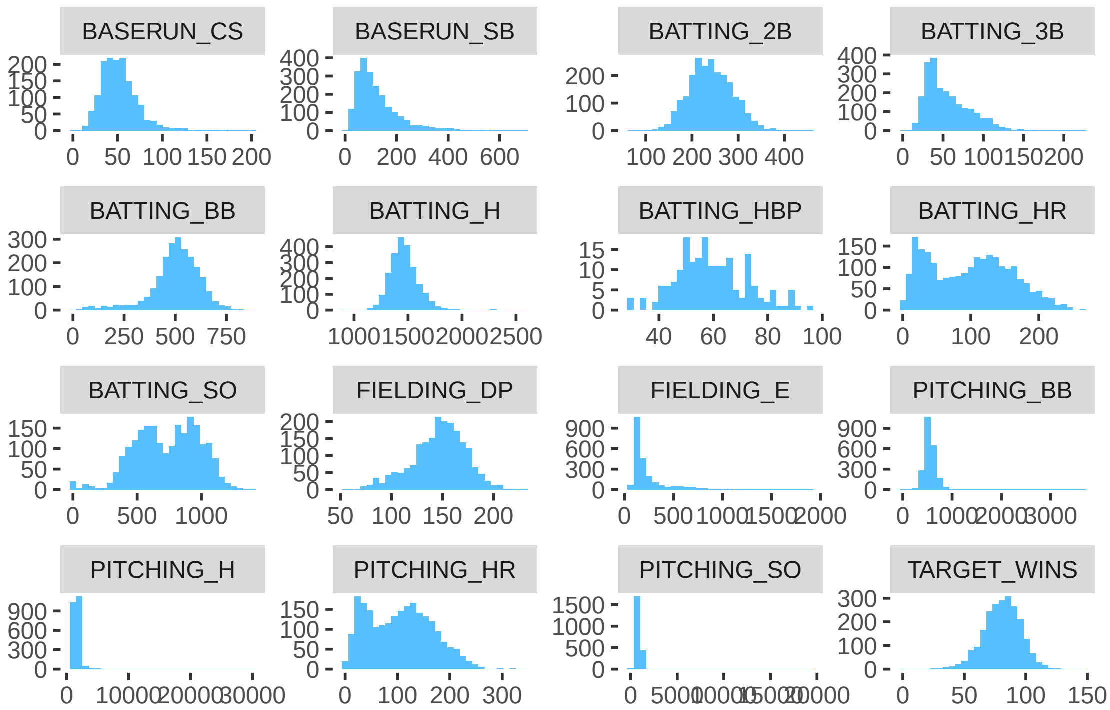


Figure 1: Data Distributions

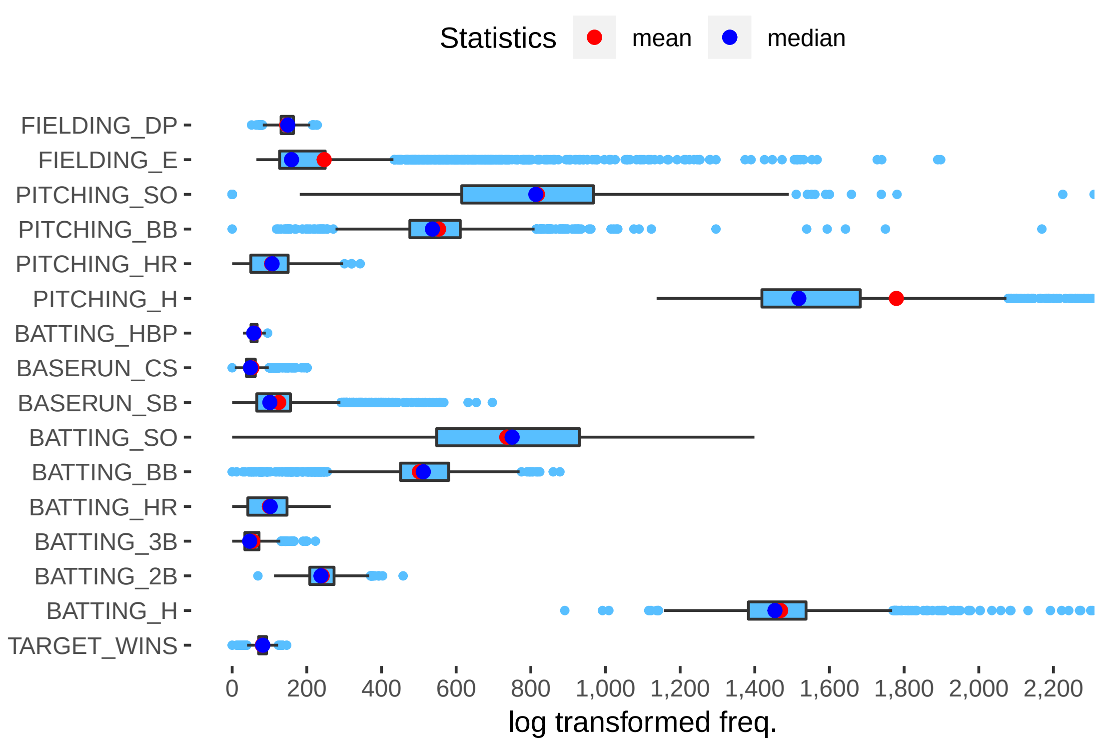


Figure 2: Boxplots highlighting many outliers in the data.

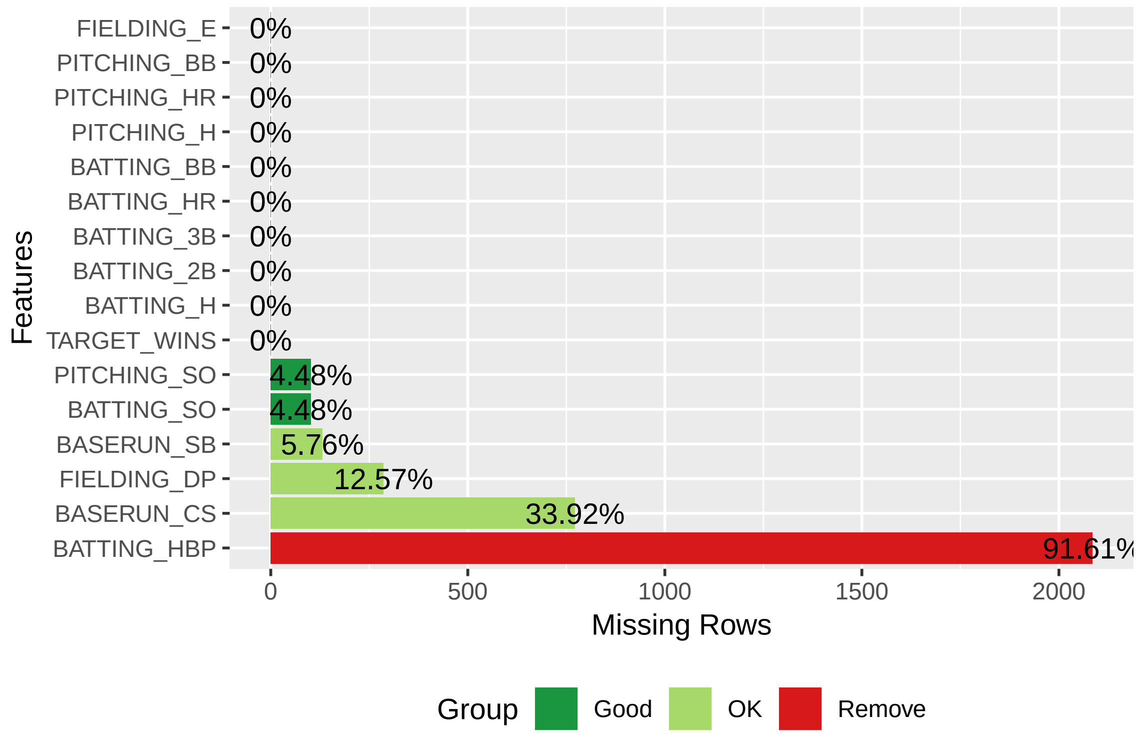


Figure 3: Missing values

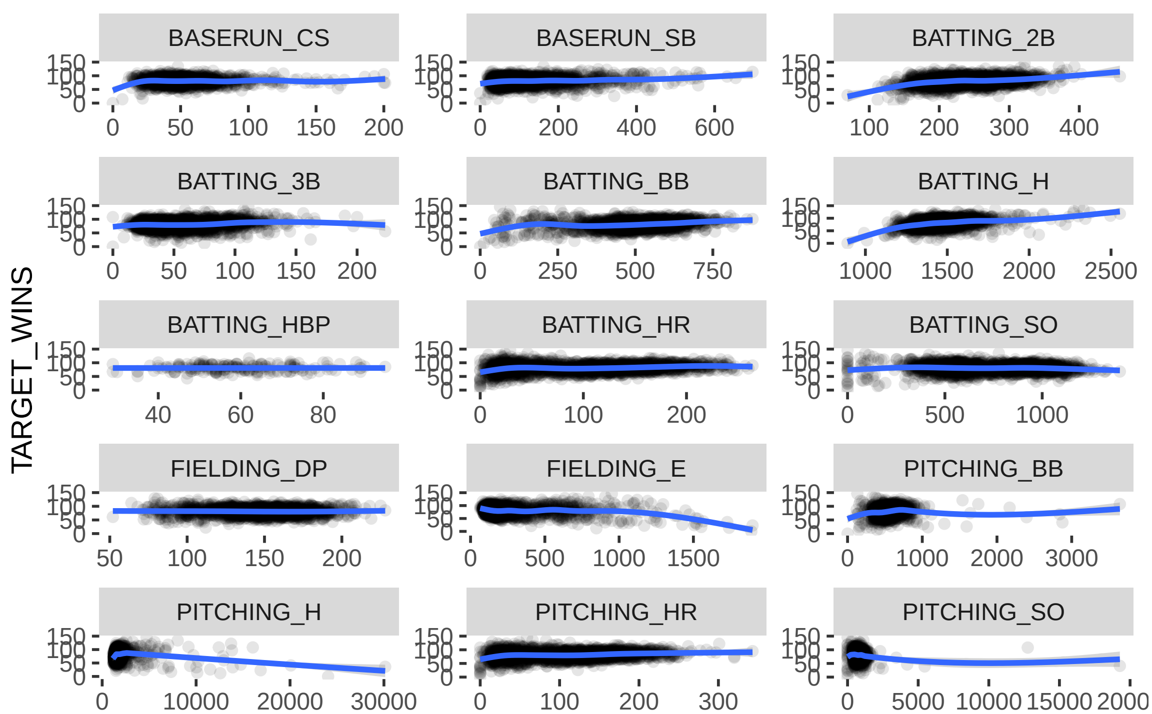


Figure 4: Linear relationships between each predictors and the target

2 DATA PREPARATION

2.1 Missing Values

As previously mentioned, just north of 10% of the data was missing values. Missing values can lead to errors in a model, bias, and worse if left unaccounted for. Attempting to “fix” this by imputing values or guessing why the values are missing in the first place - such as concluding that the missing values are meant to be zeroes - are just as likely to help with creating a model as it is to help with creating a disaster.

One of the R packages utilized, DataExplorer, which was used for the chart in the “DATA EXPLORATION” section above, recommends removing null or missing values above a certain threshold as indicated in the graph.

Fixing missing values with imputation may help, but can also have a negative impact on the model if the assumed values do not correspond to the actual missing values. When it is just a few observations missing, modifications can be made, however, 91.61% is too large a proportion and would almost definitely distort the model, so we decided it was better to remove the `BATTING_HBP` column altogether. Deleting all cases with missing values, in this instance, would have shrunk the size of the dataset down to less than a tenth of its original size. If we simply delete all cases with missing values from the analysis, we will cause no bias, but we would most certainly lose a lot of important information.

Data that is Missing Completely at Random (MCAR), meaning the probability that a value is missing is the same for all cases can be imputed. Although there is some concern about whether or not `Pitching_SO` and `Batting_SO` are MCAR, we chose to leave all the remaining variables except `BATTING_HBP` and determine whether or not to remove themn during the modelling process.

2.1.1 NA Imputation

To deal with the remaining missing values, we used preProcess via caret package. The caret package stands for Classification and regression training, it is a set of functions that streamline the process for creating predictive models, offers data splitting, pre-processing, feature selection, model tuning using resampling and variable importance estimationn.

For imputation, the selected method is bagImpute. For each predictor in the data, a bagged tree is created using all of the other predictors in the train dataset. When a new sample has a missing predictor value, the bagged model is used to predict the value. In theory, this is a more powerful method of imputing compared to KnnImpute, however, the computational cossts are much higher is the downside. We can see the difference between the original and imputed data at the Table. 3.

2.2 Remove Outliers

Outlier treament was done by placing a threshold of five times the standard deviation up from the mean and removing all observations that fell north of this boundary.

2.3 Correlation

The theoretical effect of strikeouts by batters, batters caught stealing, errors, walks, hits, and homeruns allowed were believed to have a negative impact on the number of wins of an individual team in a given year. A closer look at the correlation plot between the variables painted a different picture. Figure. 5 shows correlation plot.

Table 3: Difference between original and imputed data

	n	min	mean	median	max	sd
TARGET_WINS	0	0	0.0000000	0.00000	0	0.000000
BATTING_H	0	0	0.0000000	0.00000	0	0.000000
BATTING_2B	0	0	0.0000000	0.00000	0	0.000000
BATTING_3B	0	0	0.0000000	0.00000	0	0.000000
BATTING_HR	0	0	0.0000000	0.00000	0	0.000000
BATTING_BB	0	0	0.0000000	0.00000	0	0.000000
BATTING_SO	-102	0	7.4390885	11.00000	0	1.819230
BASERUN_SB	-131	0	-0.5740613	-4.50000	0	2.434355
BASERUN_CS	-772	0	-19.6422923	-7.98646	0	-18.178923
PITCHING_H	0	0	0.0000000	0.00000	0	0.000000
PITCHING_HR	0	0	0.0000000	0.00000	0	0.000000
PITCHING_BB	0	0	0.0000000	0.00000	0	0.000000
PITCHING_SO	-102	0	9.7505612	10.50000	0	10.133128
FIELDING_E	0	0	0.0000000	0.00000	0	0.000000
FIELDING_DP	-286	0	1.1365415	3.00000	0	1.378536

When compared to what was hypothesized, there was actually a positive impact for the number of wins for a team in a given year by walks, hits, and homeruns allowed; at the same time, variables previously thought to have a positive correlation - strikeouts by pitchers and double plays - had a negative correlation for the number of wins. The three variables with the greatest correlation to the number of wins were the hits allowed, the walks by batters, and the walks allowed. Of these, the hits allowed had a relatively low correlation with the walks by batters and the walks allowed, whereas the walks allowed and the walks by batters had a direct positive correlation with one another.

2.4 Feature Engineering

Since there are four pairs of related variables that are two sides of the same coin, hits allowed vs. hits by batters, home runs allowed vs. home runs hit by batters, etc. and three of those pairs are highly correlated with each other we decided to try using the difference between them in place of the original variables in our original models. We decided to use offense (batting) minus defense (pitching). These arithmetically transformed offense / defense variables are linearly related with BATTING and PITCHING variables, so we can include one or the other in a model, but not both. However, replacing original variables with these transforms did not improve R^2 in a base case.

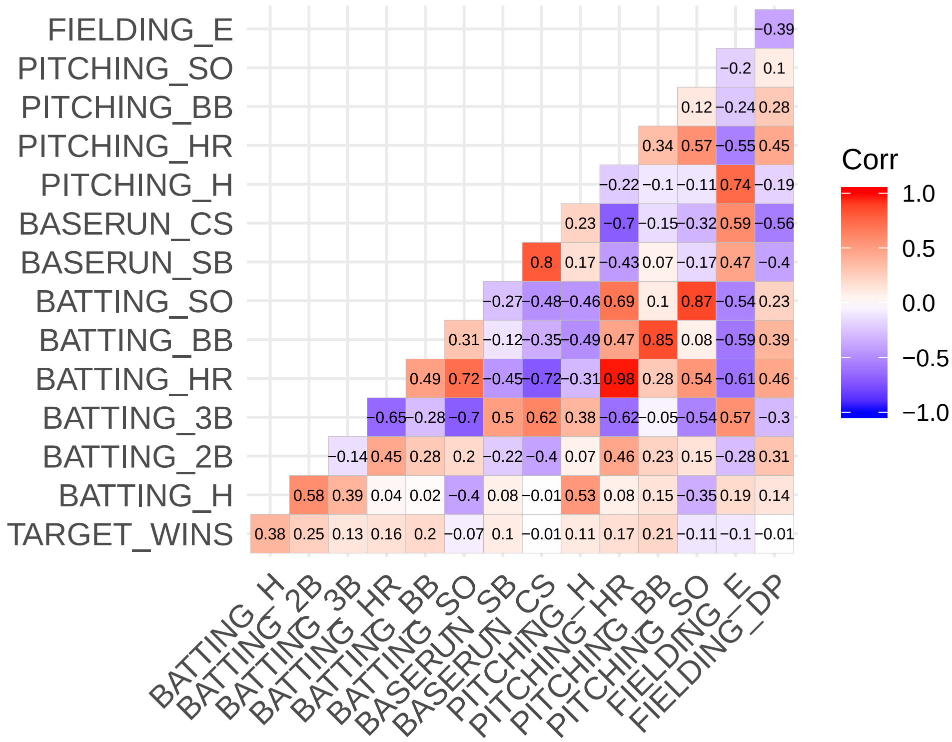


Figure 5: Correlation

---

3 BUILD MODELS

3.1 MODEL 1

Multiple regression can be created as a purely statistical model through the use of significance tests, or it can be interpreted in a more practical, non-statistical manner. This first approach is more of the latter and is based on consultation with a subject-area expert.

After discussing the various valuables available with the expert, the following categories from the most important to the least important variables were developed:

Very Important: BATTING_H, BATTING_HR, BATTING_SO, FIELDING_E, PITCHING_SO

Fairly Important: BASERUN_SB, PITCHING_HR, BATTING_BB

Important: BATTING_2B, BATTING_3B, FIELDING_DP, PITCHING_H

Slightly Important: PITCHING_BB, BASERUN_CS

Not Important: BATTING_HBP

Reviewing this preliminary categorization led to the expert suggesting the elimination of `BATTING_HBP` and `BASERUN_CS` as they are the least important when determining the number of wins in a season as per their professional opinion.

```
##                Estimate  Std. Error     t value     Pr(>|t|)
## (Intercept) 37.10977143 5.653204621   6.5643779 6.487024e-11
## BATTING_H    0.01685588 0.004992382   3.3763200 7.473027e-04
## BATTING_HR   0.11843578 0.050490736   2.3456934 1.907946e-02
## BATTING_SO  -0.03197607 0.006489808  -4.9271211 8.962349e-07
## FIELDING_E  -0.03924096 0.003432709 -11.4314852 1.912518e-29
## PITCHING_SO  0.02040965 0.005348821   3.8157293 1.394993e-04
## BASERUN_SB   0.04337204 0.004919369   8.8165856 2.336863e-18
## PITCHING_HR -0.03284260 0.046731061  -0.7028002 4.822541e-01
## BATTING_BB   0.08017467 0.016100593   4.9796097 6.862900e-07
## BATTING_2B  -0.01119577 0.009368451  -1.1950505 2.321952e-01
## BATTING_3B   0.12497996 0.018095211   6.9067973 6.452663e-12
## FIELDING_DP -0.09701929 0.013103882  -7.4038586 1.869009e-13
## PITCHING_BB -0.05691719 0.014263610  -3.9903773 6.811159e-05
## PITCHING_H   0.01352997 0.001534220   8.8187993 2.292665e-18
```

Creating a linear model sans the two aforementioned variables led to an Adjusted R^2 of 0.2793 on Adjusted R^2. The R^2's value decreased when attempting to remove other less-than-important variables.

While forward selection had mild success, it was determined the best next step would be to perform backwards elimination. Based on this method and its results, `BATTING_H` and `BATTING_2B` were removed.

This resulted in the following model:

```
TARGET_WINS ~ BATTING_H + BATTING_HR + BATTING_SO +
    FIELDING_E + PITCHING_SO + BASERUN_SB + BATTING_BB + BATTING_3B +
    FIELDING_DP + PITCHING_BB + PITCHING_H
```

```
##                Estimate  Std. Error     t value     Pr(>|t|)
## (Intercept) 51.62624026 3.805048265  13.5678280 2.415588e-40
## BATTING_HR   0.13634248 0.050331310   2.7088999 6.802559e-03
## BATTING_SO  -0.03381764 0.006478440  -5.2200291 1.955523e-07
## FIELDING_E  -0.04266213 0.003245216 -13.1461610 4.565173e-38
## PITCHING_SO  0.01891741 0.005305604   3.5655519 3.707738e-04
## BASERUN_SB   0.04771459 0.004774412   9.9938153 4.940051e-23
## PITCHING_HR -0.03617662 0.046825149  -0.7725896 4.398476e-01
## BATTING_BB   0.08760025 0.015942379   5.4948040 4.359717e-08
## BATTING_3B   0.15521913 0.015997020   9.7030027 7.897355e-22
## FIELDING_DP -0.08953847 0.012952873  -6.9126337 6.196508e-12
## PITCHING_BB -0.06303832 0.014157328  -4.4526992 8.900393e-06
## PITCHING_H   0.01697833 0.001180506  14.3822404 6.567546e-45
```

The R^2 produced was still low (0.276), so outliers were analyzed as more thoroughly due to the effect they can have on models. `PITCHING_H` had a high number of outliers which indicated a need for data transformation; log transformation was used to normalize this variable and further bring the linear model in line.

```
##                      Estimate   Std. Error     t value     Pr(>|t|)
## (Intercept)     -3.016025e+02 24.123697275 -12.5023311 1.041483e-34
## BATTING_HR       1.132170e-01  0.049894396   2.2691333 2.335589e-02
## BATTING_SO      -1.667687e-02  0.006559479  -2.5424072 1.107664e-02
## FIELDING_E      -4.293805e-02  0.003160236 -13.5869769 1.897644e-40
## PITCHING_SO      8.351735e-03  0.005257322   1.5885912 1.122952e-01
## BASERUN_SB       4.335272e-02  0.004671241   9.2807718 3.874552e-20
## PITCHING_HR     -4.153052e-02  0.046276541  -0.8974421 3.695805e-01
```

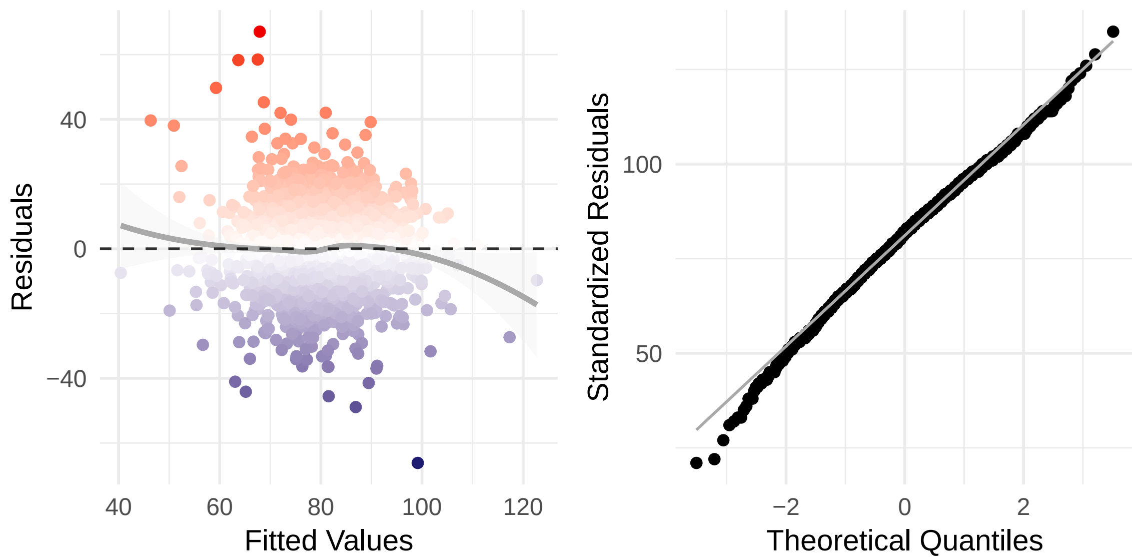


Figure 6: Model 1: Residual Plot and Q-Q Plot

```
## BATTING_BB        1.028744e-01  0.015906293   6.4675311 1.221619e-10
## BATTING_3B        1.156954e-01  0.015898209   7.2772570 4.705050e-13
## FIELDING_DP      -9.974519e-02  0.012830583  -7.7740190 1.154549e-14
## PITCHING_BB      -7.679186e-02  0.014135932  -5.4323870 6.168839e-08
## log(PITCHING_H)   5.203154e+01  3.224550721  16.1360598 1.807885e-55
```

After we used the log transformation the model's Adjusted R^2 increased to 0.2921.

Residuals for this linear model are normally distributed and random. The Q-Q plot confirms this model can be used to predict the number of wins in a season for a team, though there is still room for improvement using other methods.

Below is Model 1's prediction result for the test data:

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.    NA's 
##   60.71   75.87   80.98   81.41   86.10  108.62      54
```

3.1.1 Summary of Results

r.squared	adj.r.squared	statistic	p.value	df	deviance	df.residual
0.2996014	0.2961278	86.25164	0	12	345591.5	2218

It was determined the overall subject-area expertise wasn't as effective as a stand-alone method of creating multiple regression models. Statistical iterations which were performed contradicted the subject area expert such as removing `BATTING_H` from the model. Additionally the log tranformation of `PITCHING_H` made a significant improvement in the model's linearity.

3.2 MODEL 2

Our approach for Model 2 was to try to use as many of the tools as possible that are available in R and that we have learned thus far to determine a model based solely on the statistical qualities of the predictor

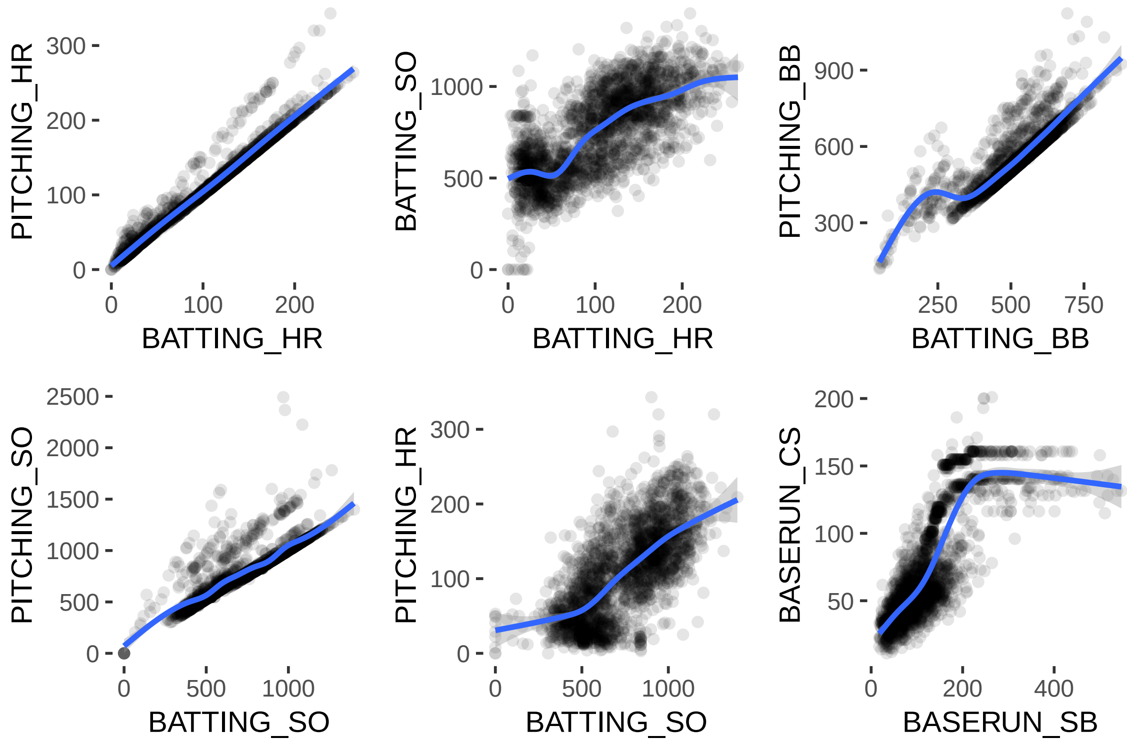


Figure 7: Scatterplots showing possible collinearity problems

variables without any regard to our expert's opinion.

3.2.1 Check for Correlated Predictor Variables and Linear Relationship to Target

We started by plotting the relationships between variables that had high correlation values to look for potential collinearity problems.

Based on the charts above we decided somewhat arbitrarily to remove the three pitching variables (`PITCHING_HR`, `PITCHING_BB`, and `PITCHING_SO`) rather than the corresponding batting variables (`BATTING_HR`, `BATTING_BB`, and `BATTING_SO`) due to the extremely high correlation between these predictors.

We then plotted the remaining variables to see if they showed a linear relationship with the target variable. Most of the premaining predictors showed a clear linear relationship with the target, however, the extreme skew of `PITCHING_H` and `FIELDING_E` as well as a more moderate skew in `BASERUN_SB` and `BATTING_3B`, can be seen in the plots.

3.2.2 Log Transform Data

We decided to log transform `PITCHING_H`, `FIELDING_E`, `BASERUN_SB` and `BATTING_3B` in order to compensate for the skew. The resulting distributions can be seen in the revised plots below.

3.2.3 Building the First Model

Finally we built a model based on the selected variables including the log transformations where appropriate.

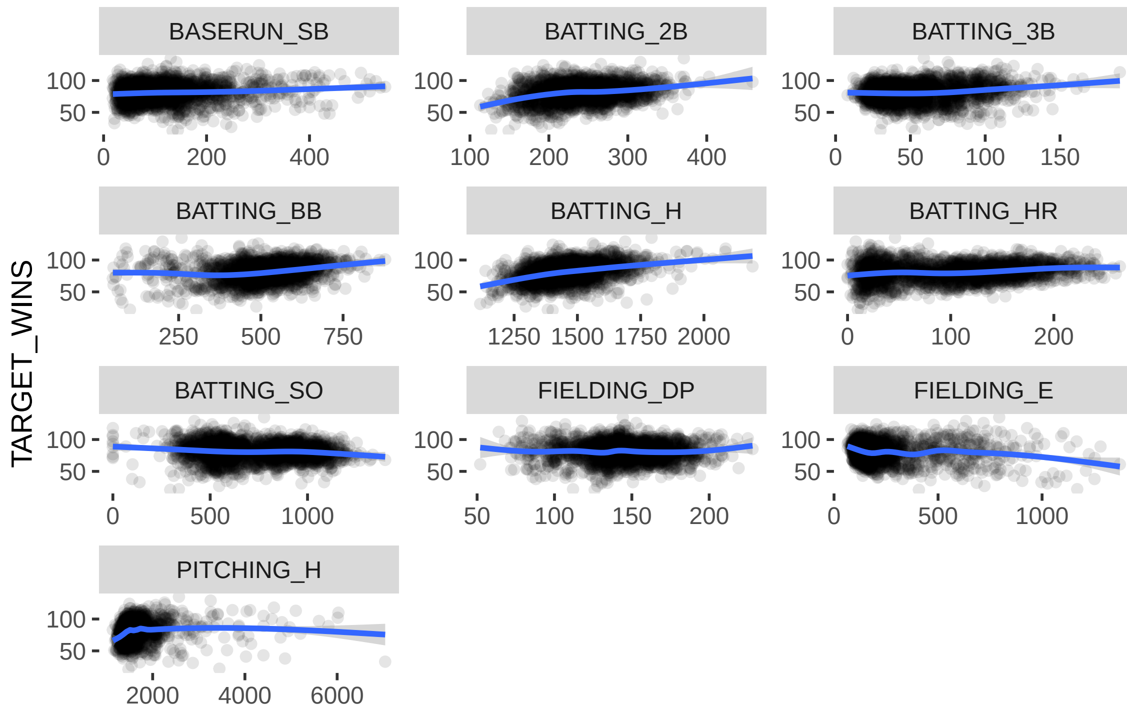


Figure 8: Linear relationship between each predictor and the target showing highly skewed variables

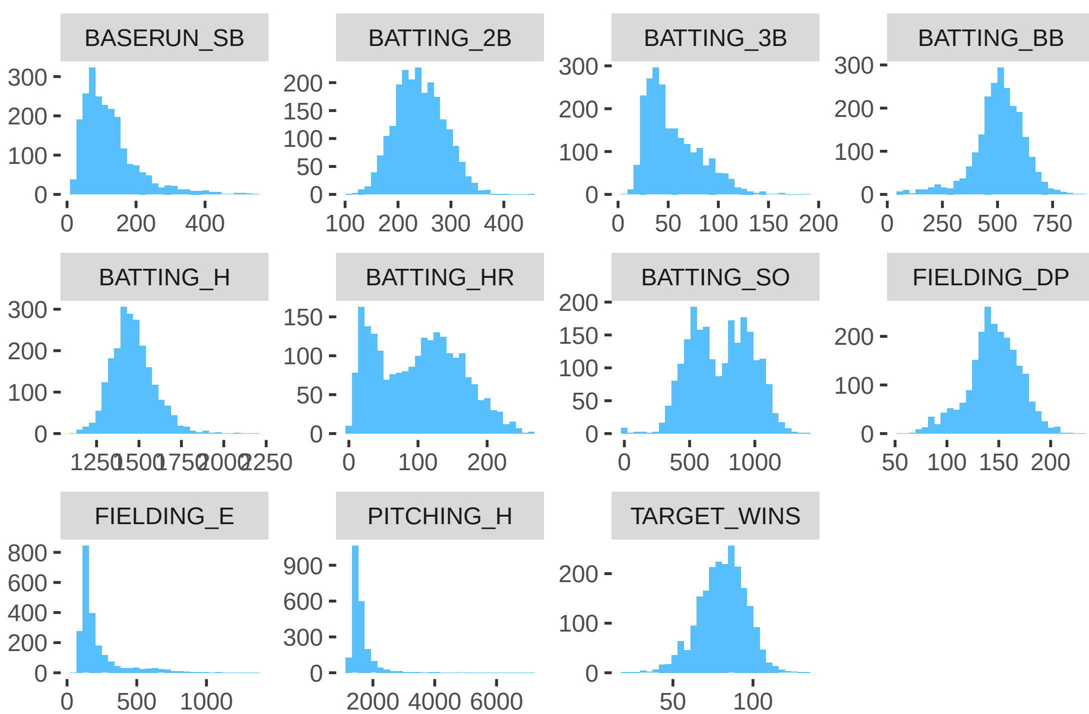


Figure 9: Predictor variable distributions showing highly skewed variables

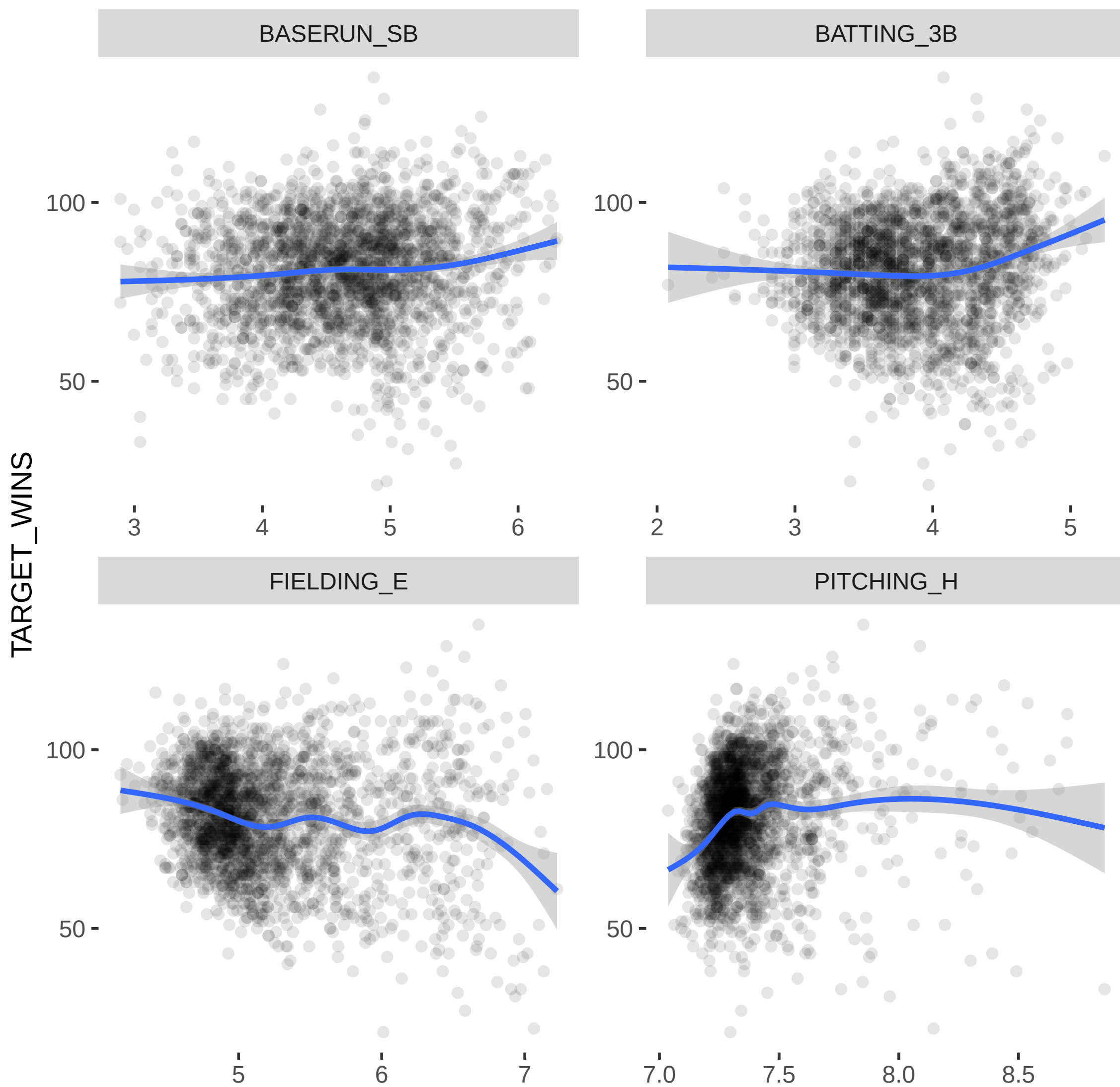


Figure 10: Linear relationship between each log transformed predictor and the Target showing decreased skew

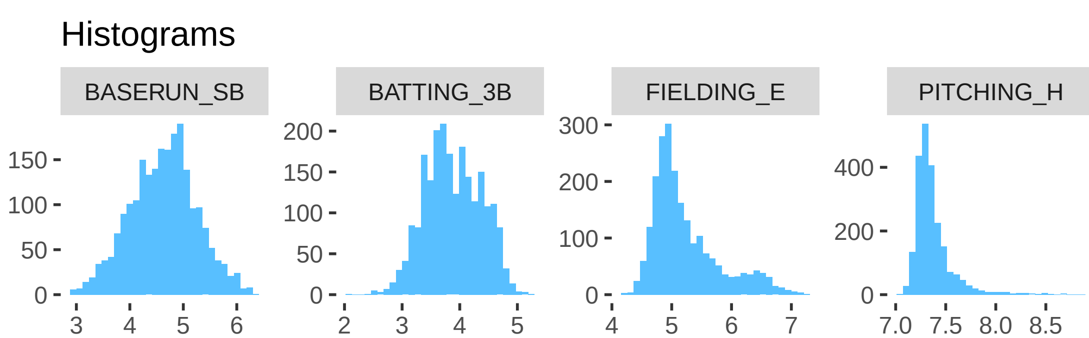


Figure 11: Log transformed distributions showing decreased skew

Table 4: First Model Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-59.2869532	15.8583253	-3.738538	0.0001898
BATTING_H	0.0257445	0.0045887	5.610429	0.0000000
BATTING_2B	-0.0297056	0.0091911	-3.232011	0.0012472
log(BATTING_3B)	6.9587093	0.9453783	7.360767	0.0000000
BATTING_HR	0.0715758	0.0100980	7.088118	0.0000000
BATTING_BB	0.0198950	0.0031811	6.254201	0.0000000
BATTING_SO	-0.0116432	0.0024106	-4.830048	0.0000015
log(BASERUN_SB)	4.7111493	0.5621718	8.380266	0.0000000
log(PITCHING_H)	18.7049288	2.6055575	7.178859	0.0000000
log(FIELDING_E)	-13.2665493	1.0609676	-12.504198	0.0000000
FIELDING_DP	-0.1122291	0.0131625	-8.526455	0.0000000

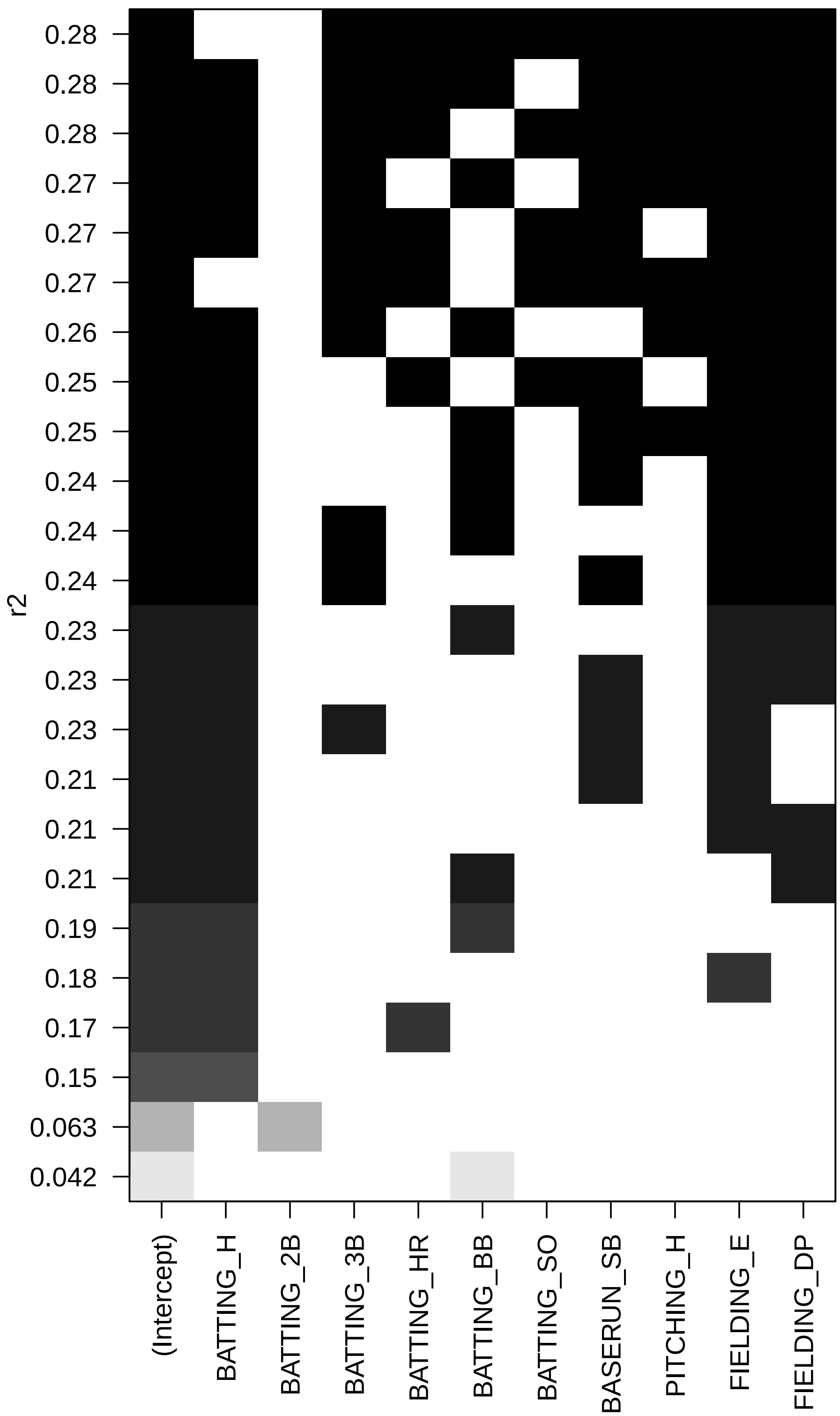
```
TARGET_WINS ~ BATTING_H + BATTING_2B + log(BATTING_3B) + BATTING_HR +
    BATTING_BB + BATTING_SO + log(BASERUN_SB) + log(PITCHING_H) +
    log(FIELDING_E) + FIELDING_DP
```

All of the variables had a very low p-value indicating a significant impact on our target, however our R^2 value was low at only 0.2889.

3.2.3.1 First Model R^2 0.2926739

3.2.4 Refining the Model with leaps Package

We thought we may be able to use some other tools in R to refine our model and get a better R^2 value. So next we tried using the leaps package to see if it would recommend removing any of our chosen variables from the model. In the following plot you can see that we could remove `BATTING_H`, `BATTING_2B` without affecting out R^2 much, but it would not improve the model.


r2
0.28
0.28
0.28
0.27
0.27
0.27
0.26
0.25
0.25
0.24
0.24
0.24
0.23
0.23
0.23
0.21
0.21
0.21
0.19
0.18
0.17
0.15
0.063
0.042
(Intercept)
BATTING_H
BATTING_2B
BATTING_3B
BATTING_HR
BATTING_BB
BATTING_SO
BASERUN_SB
PITCHING_H
FIELDING_E
FIELDING_DP

```
## NULL
```

3.2.5 Refining the Model by Standardizing the Predictor Variables

Next we tried standardizing the (non-log-transformed) variables to see what impact that might have on our model. Standardizing actually resulted in a significant reduction in our R^2 value from 0.2889 to 0.274.

3.2.5.1 STANDARDIZED Model R^2 0.2783138

3.2.6 Test all of the predictors

Next we ran an ANOVA test to compare our model to the null model. With a p-value that is basically zero, clearly our model is statistically significant.

Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
2229	493421.2	NA	NA	NA	NA
2219	349009.7	10	144411.5	91.81668	0

3.2.7 Testing a subspace

We then tried testing a subspace. Since our initial models using the difference between the corresponding batting and pitching variables did not show promise we tried adding those two variables instead.

Once again our model declined in performance rather than improving.

3.2.7.1 Subspace Model R^2 0.289544

3.2.8 Refining the Model with the MASS Package

Last, but not least, we used the stepAIC function from the MASS package to see if it came up with different recommendations for what variabels to keep and which to exclude from our model. We started with all variables putting back the ones we had previously taken out due to collinearity issues and let the algorithm choose which to keep.

The final suggested model was:

```
Final Model:
TARGET_WINS ~ BATTING_H + BATTING_3B + BATTING_HR + BATTING_BB +
    BATTING_SO + BASERUN_SB + BASERUN_CS + PITCHING_H + PITCHING_BB +
    PITCHING_SO + FIELDING_E + FIELDING_DP
```

In comparison to our original model we had the following variables added to our model (`BASERUN_CS`, `PITCHING_BB`, and `PITCHING_SO`) and the following variable removed (`BATTING_2B`).

We tried multiple iterations of that model, without any log transformations, with log transformations, with and without the collinear variables, but whever we removed one of the collinear variables our model would decline in performance, so we decided to try our multiplying the corresponding collinear variables to gether and BINGO! We got and R^2 of 0.3247 using the following model:

```
TARGET_WINS ~ BATTING_3B + BATTING_HR + BATTING_BB*PITCHING_BB +
    BATTING_SO*PITCHING_SO + BASERUN_SB + BASERUN_CS + BATTING_H*log(PITCHING_H) +
    log(FIELDING_E) + FIELDING_DP
```

Table 5: FINAL Model 2 Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	206.9341788	102.7932696	2.0131102	0.0442239
BATTING_3B	0.1575555	0.0178605	8.8214372	0.0000000
BATTING_HR	0.0708461	0.0100892	7.0219925	0.0000000
BATTING_BB	0.0646559	0.0199004	3.2489710	0.0011756
PITCHING_BB	-0.0873147	0.0129172	-6.7595805	0.0000000
BATTING_SO	0.0009820	0.0090401	0.1086294	0.9135063
PITCHING_SO	0.0291645	0.0063855	4.5673009	0.0000052
BASERUN_SB	0.0324383	0.0062552	5.1858348	0.0000002
BASERUN_CS	0.0525705	0.0154361	3.4056789	0.0006718
BATTING_H	-0.2191615	0.0539942	-4.0589783	0.0000510
log(PITCHING_H)	-8.2256915	14.1969972	-0.5793966	0.5623806
log(FIELDING_E)	-16.7290462	1.1110683	-15.0567210	0.0000000
FIELDING_DP	-0.0964320	0.0131909	-7.3105034	0.0000000
BATTING_BB:PITCHING_BB	0.0000459	0.0000168	2.7256225	0.0064686
BATTING_SO:PITCHING_SO	-0.0000273	0.0000044	-6.2665062	0.0000000
BATTING_H:log(PITCHING_H)	0.0304119	0.0073681	4.1275360	0.0000380

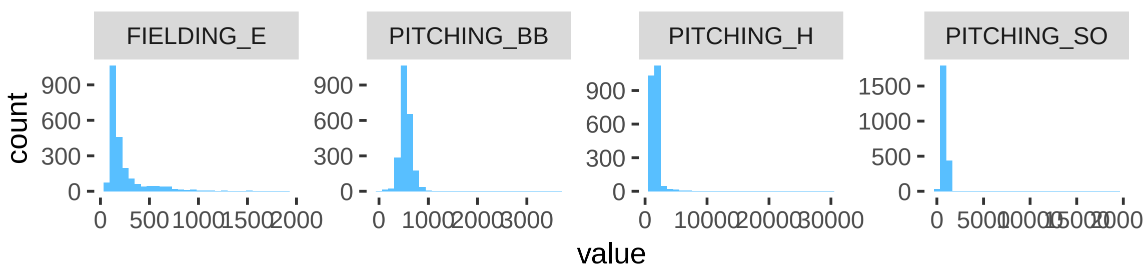


Figure 12: Histograms of variables showing pronounced rightward-skew

3.2.8.1 FINAL Model 2 R^2 0.3275118

3.3 MODEL 3

We sought to explore whether there was a relationship between wins and the difference of specific offensive and defensive team capabilities - hits, homeruns, balls, and strike-outs. Incorporating variables that reflect those differences (i.e. subtracting batting hits from pitching hits, and so on), however, did not improve the explantory power of the model beyond using the original variables.

Given these variables did not yield improvements, in their place we explored a third model. As the histograms below highlight, a number of the independent variables - pitching hits, pitching homeruns, pitching strikeouts - demonstrate pronounced rightward-skew.

We corrected for that skew by transforming those three variables using natural logarithms. When we tested those log transformations in a model where they replaced the untransformed, original variables combined with all other variables, we found that neither the originals nor the log transformations for pitching homeruns and pitching strikeouts met the threshold of significance (a p-value below the α level of .05). Based on high p-values, over a series of backward steps we removed pitching homeruns, pitching strikeouts, and baserun caught stealing, yielding the following model:

Table 6: Log Transform Model Coefficients

	Estimate	Std. Error	t value	Pr(>|t|)
(Intercept)	-401.0200852	36.8354457	-10.886799	0.0000000
BATTING_H	-0.0131785	0.0060037	-2.195065	0.0282622
BATTING_2B	-0.0148208	0.0091202	-1.625049	0.1042942
BATTING_3B	0.1384812	0.0176241	7.857502	0.0000000
BATTING_HR	0.0738318	0.0096827	7.625132	0.0000000
BATTING_BB	0.1243185	0.0126227	9.848817	0.0000000
BATTING_SO	-0.0070344	0.0023288	-3.020625	0.0025513
BASERUN_SB	0.0456456	0.0047753	9.558759	0.0000000
log(PITCHING_H)	68.1753429	5.7326910	11.892381	0.0000000
PITCHING_BB	-0.0945309	0.0105985	-8.919258	0.0000000
FIELDING_E	-0.0482011	0.0035562	-13.554108	0.0000000
FIELDING_DP	-0.0939709	0.0129489	-7.257048	0.0000000

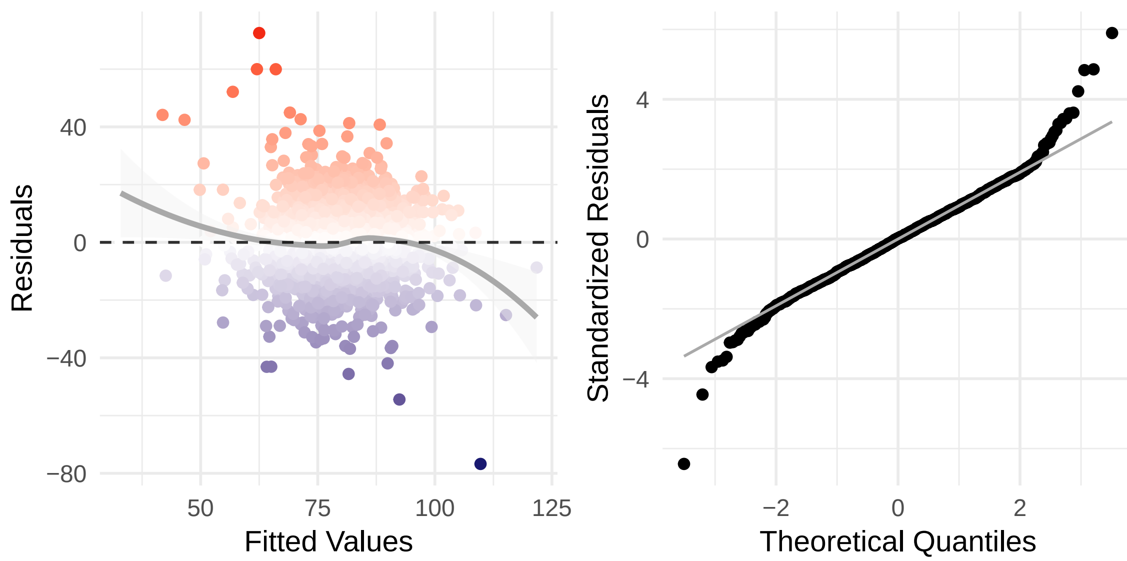


Figure 13: Model 3: Residual Plot and Q-Q Plot

3.3.1 Residual and Q-Q Plot

Based on this model's F-statistic and p-value, we can reject the null hypothesis that coefficients with values of zero would fit the data better. Per the adjusted r^2 value, this model explains approximately 29.56% of the variance in wins. However, in doing so it treats the batting hits and batting second base runs as drags on wins (with negative coefficients), and pitching hits as buoying wins - which is counterunituitive. While the other coefficients make more intuitive sense, these signs call into question how effectively we can use this model to understand the relationships between the independent variables and wins.

4 SELECT MODELS

4.1 Instructions:

Decide on the criteria for selecting the best multiple linear regression model. Will you select a model with slightly worse performance if it makes more sense or is more parsimonious? Discuss why you selected your model. For the multiple linear regression model, will you use a metric such as Adjusted R2, RMSE, etc.? Be sure to explain how you can make inferences from the model, discuss multi-collinearity issues (if any), and discuss other relevant model output. Using the training data set, evaluate the multiple linear regression model based on (a) mean squared error, (b) R2, (c) F-statistic, and (d) residual plots. Make predictions using the evaluation data set.

4.2 Comparison of models

In order to determine which model was best suited for determining the number of wins in a season of the three developed, two factors were considered primarily. First and foremost was the R^2 of the models - Model 2 had the highest with an R^2 of 0.3247, while Model 1 had an R^2 of 0.2996 and Model 3 had an R^2 of 0.2956. Secondarily, the coefficients were considered as to whether or not they made sense - the first and second model's both did, whereas the third's did not due to having negative values. It was these two factors that led to the second model ultimately being chosen.

4.3 Multi-collinearity

An examination of the partial correlation coefficients between all the independent variables shows that the expected relationships between the offense and defense variables - i.e. batting and pitching homeruns - are there, meet p-value thresholds, and are strong. This finding suggests keeping only one of each pair of variables in a given model; yet, when we tried this, less of the variance in wins was explained by the model. As a result we chose to transform each set of two collinear variables by multiplying them together into one combined variable in our model. This resulted in the highest R^2 value of any of the models we created.

4.4 Check Conditions for Least Squares Regression

Our final model's residuals are normally distributed and random. Furthermore, the residuals present with constant variability and no indication of homoscedasticity.

4.5 Predictions

We ran predictions on our final model and plotted the distribution next to the distribution from our target in the training data set to compare…

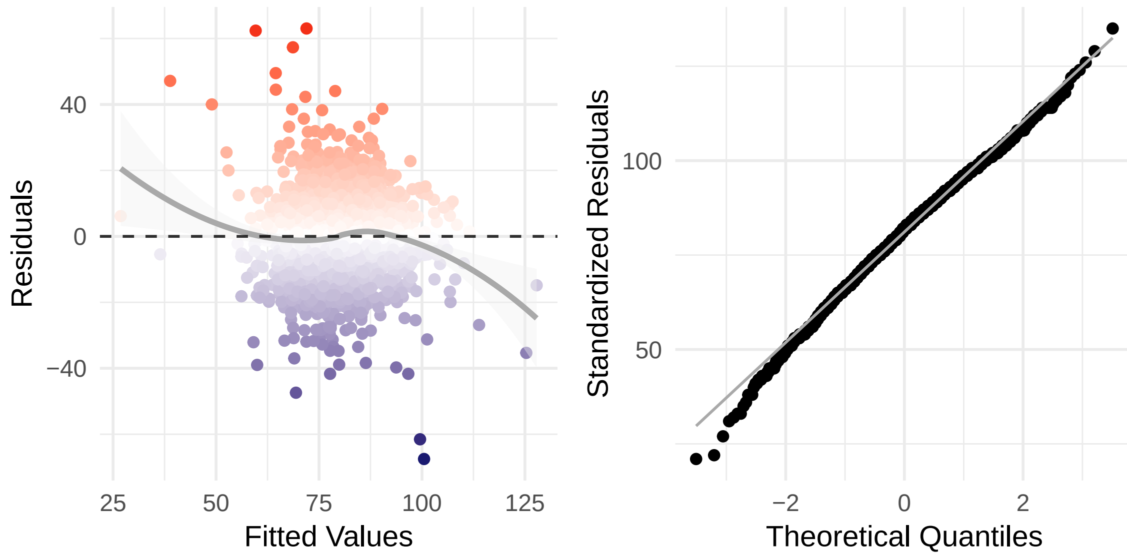


Figure 14: Model 2: Residual Plot and Q-Q Plot

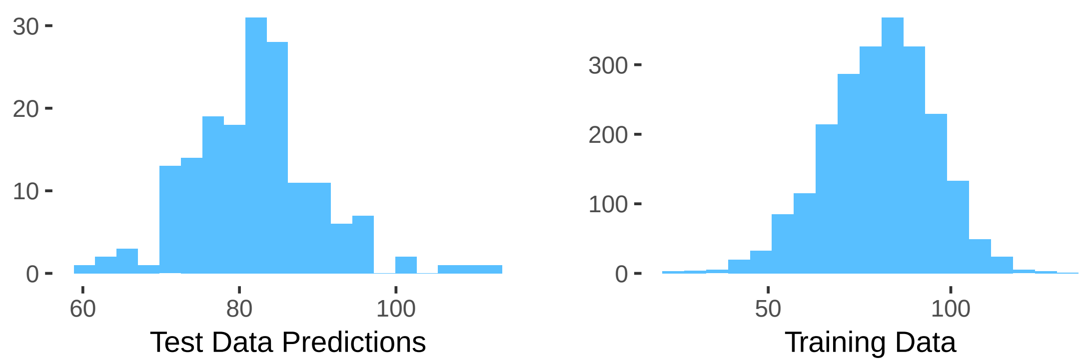


Figure 15: Predictions vs. training data

5 Appendix